CLAIMS

What is claimed is:

1. An apparatus for performing correctness checks opportunistically, the apparatus comprising:

first logic, the first logic receiving a first set of instructions and generating an initial instruction schedule from the first set of instructions, the first set of instructions including one or more instructions associated with a correctness check function;

second logic, the second logic evaluating the initial instruction schedule to determine whether the initial instruction schedule includes spare instruction slots into which said one or more instructions associated with the correctness check function can be inserted; and

third logic, the third logic inserting said one or more instructions associated with the correctness check function into the spare instruction slots if enough spare instruction slots exist in the initial instruction schedule for accommodating said one or more instructions.

2. The apparatus of claim 1, wherein said one or more instructions associated with the correctness check function correspond to a conditional expression, and wherein the first logic performs initial code generation prior to generating the initial instruction schedule, wherein when the first logic performs initial code generation, said one or more instructions associated with the correctness check function are separated from all other instructions of said first set of instructions so that the initial instruction schedule does not include any instructions associated with the correctness check function.

3. The apparatus of claim 2, wherein said first, second and third logic correspond to a processor programmed to execute a compiler program, the compiler program including a first code segment for performing initial code generation and for generating the initial instruction schedule, a second code segment for evaluating the initial instruction schedule to determine whether spare instruction slots exist in the initial instruction schedule, and a third code segment for inserting said one or more instructions associated with the correctness check function into the spare instruction slots if enough spare instruction slots exist to accommodate said one or more instructions.

4. An apparatus for performing correctness checks opportunistically, the apparatus comprising:

first means for receiving a first set of instructions and for generating an initial instruction schedule from the first set of instructions, the first set of instructions including one or more instructions associated with a correctness check function;

second means for evaluating the initial instruction schedule to determine whether the initial instruction schedule includes spare instruction slots into which said one or more instructions associated with the correctness check function can be inserted; and

third means for inserting said one or more instructions associated with the correctness check function into the spare instruction slots if enough spare instruction slots exist in the initial instruction schedule for accommodating said one or more instructions.

5. The apparatus of claim 4, wherein said one or more instructions associated with the correctness check function correspond to a conditional expression, and wherein the first means performs initial code generation prior to generating the initial instruction schedule, wherein when the first logic performs initial code generation, said one or more instructions associated with the correctness check function are separated from all other instructions of said first set of instructions so that the initial instruction schedule does not include any instructions associated with the correctness check function.

6. A method for performing correctness checks opportunistically, the method comprising the steps of:

receiving a first set of instructions and generating an initial instruction schedule from the first set of instructions, the first set of instructions including one or more instructions associated with a correctness check function;

evaluating the initial instruction schedule to determine whether the initial instruction schedule includes spare instruction slots into which said one or more instructions associated with the correctness check function can be inserted; and

inserting said one or more instructions associated with the correctness check function into the spare instruction slots if enough spare instruction slots exist in the initial instruction schedule for accommodating said one or more instructions.

7. The method of claim 6, wherein said one or more instructions associated with the correctness check function correspond to a conditional expression, and wherein the step of generating the initial instruction schedule includes the step of performing initial code generation, wherein when initial code generation is performed, said one or

more instructions associated with the correctness check function are separated from all other instructions of said first set of instructions so that the initial instruction schedule does not include any instructions associated with the correctness check function.

8. The method of claim 7, wherein the method is performed by a processor programmed to execute a compiler program, the compiler program including a first code segment for performing initial code generation and for generating the initial instruction schedule, a second code segment for evaluating the initial instruction schedule to determine whether spare instruction slots exist in the initial instruction schedule, and a third code segment for inserting said one or more instructions associated with the correctness check function into the spare instruction slots if enough spare instruction slots exist to accommodate said one or more instructions.

9. A computer program for performing correctness checks opportunistically, the computer program being embodied on a computer-readable medium, the computer program comprising:

a first code segment, the first code segment generating an initial instruction schedule from a first set of instructions, the first set of instructions including one or more instructions associated with a correctness check function;

a second code segment, the second code segment evaluating the initial instruction schedule to determine whether the initial instruction schedule includes spare instruction slots into which said one or more instructions associated with the correctness check function can be inserted; and

a third code segment, the third code segment inserting said one or more instructions associated with the correctness check function into the spare instruction slots if enough spare instruction slots exist in the initial instruction schedule to accommodate said one or more instructions.

10. The computer program of claim 9, wherein said one or more instructions associated with the correctness check function correspond to a conditional expression, and wherein prior to generating the initial instruction schedule, the first code segment performs initial code generation, wherein when initial code generation is performed, said one or more instructions associated with the correctness check function are separated from all other instructions of said first set of instructions so that the initial instruction schedule does not include any instructions associated with the correctness check function.